agriculture

Article

Efficient Tobacco Pest Detection in Complex Environments Using an Enhanced YOLOv8 Model

Daozong Sun [1], Kai Zhang [1], Hongsheng Zhong [1], Jiaxing Xie [1], Xiuyun Xue [1], Mali Yan [1], Weibin Wu [2] and Jiehao Li [1,2,*]

[1] College of Electronic Engineering (College of Artificial Intelligence), South China Agricultural University, Guangzhou 510642, China; sundaozong@scau.edu.cn (D.S.); zhangkai_scau@stu.scau.edu.cn (K.Z.); zhonghs@stu.scau.edu.cn (H.Z.); xjx1998@scau.edu.cn (J.X.); xuexiuyun@scau.edu.cn (X.X.); yanmali@scau.edu.cn (M.Y.)

[2] College of Engineering, South China Agricultural University, Guangzhou 510642, China; wuweibin@scau.edu.cn

* Correspondence: jiehao.li@scau.edu.cn

Abstract: Due to the challenges of pest detection in complex environments, this research introduces a lightweight network for tobacco pest identification leveraging enhancements in YOLOv8 technology. Using YOLOv8 large (YOLOv8l) as the base, the neck layer of the original network is replaced with an asymptotic feature pyramid network (AFPN) network to reduce model parameters. A SimAM attention mechanism, which does not require additional parameters, is incorporated to improve the model's ability to extract features. The backbone network's C2f model is replaced with the VoV-GSCSP module to reduce the model's computational requirements. Experiments show the improved YOLOv8 model achieves high overall performance. Compared to the original model, model parameters and GFLOPs are reduced by 52.66% and 19.9%, respectively, while mAP@0.5 is improved by 1%, recall by 2.7%, and precision by 2.4%. Further comparison with popular detection models YOLOv5 medium (YOLOv5m), YOLOv6 medium (YOLOv6m), and YOLOv8 medium (YOLOv8m) shows the improved model has the highest detection accuracy and lightest parameters for detecting four common tobacco pests, with optimal overall performance. The improved YOLOv8 detection model proposed facilitates precise, instantaneous pest detection and recognition for tobacco and other crops, securing high-accuracy, comprehensive pest identification.

Keywords: pest recognition; object detection; YOLOv8; lightweight network; attention mechanism

check for updates

Citation: Sun, D.; Zhang, K.; Zhong, H.; Xie, J.; Xue, X.; Yan, M.; Wu, W.; Li, J. Efficient Tobacco Pest Detection in Complex Environments Using an Enhanced YOLOv8 Model. *Agriculture* **2024**, *14*, 353. https://doi.org/10.3390/agriculture14030353

Academic Editor: Marco Valerio Rossi Stacconi

Received: 12 January 2024
Revised: 17 February 2024
Accepted: 20 February 2024
Published: 22 February 2024

1. Introduction

As one of the most important economic crops in China, tobacco is a key raw material for the cigarette industry. In Guangdong Province, China, tobacco is cultivated on a large scale and generates high economic outputs, thus playing a vital role in promoting local economic development [1,2]. Owing to external growth environments and its own chemical properties, tobacco is susceptible to infestation by some specialist insect pests, especially on the leaves and rootstalk. This can lead to reduced yield and quality, resulting in tremendous economic losses to the tobacco industry [3].

In the past few years, as computer vision and deep learning technologies have advanced in tandem, a growing number of deep learning approaches have been widely adopted in agricultural production. These methods have found successful applications in areas such as pest detection, crop classification, and fruit counting. These deep learning approaches autonomously extract features from image data, and their detection accuracy surpasses conventional visual inspection and machine learning methods. Moreover, the developed models are capable of being transferred to portable or integrated systems [4–6]. For instance, Liu et al. [7] utilized an enhanced AlexNet architecture for the classification of twelve prevalent species of pests in rice fields, attaining an average detection accuracy (mAP) of 95.1% on their test set. Wang et al. enhanced the accuracy to 91% in classifying crop pest images by adjusting the convolutional kernel size of AlexNet, demonstrating superiority over traditional machine learning techniques [8]. Furthermore, Cheng et al. utilized the ResNet architecture to classify 98 agricultural pest species. Their results indicated that networks based on deep residual learning structures outperformed conventional deep convolutional neural networks like AlexNet, especially in recognizing pests against complex backgrounds [9]. Collectively, these research efforts have delved into the use of deep learning in crop pest recognition, confirming the viable application of deep learning-based computer vision methods for detecting agricultural pests.

With the rapid development of deep learning, leveraging image detection and localization for pest identification has become feasible, characterized by its accuracy, real-time performance, and non-destructive nature. The methods involved primarily fall into two categories: two-stage and one-stage object detection algorithms. Two-stage algorithms, represented by models such as R-CNN [10] and Faster R-CNN [11], first obtain candidate regions of the detection object, then generate candidate bounding boxes within these areas for regression prediction, resulting in the final prediction boxes. On the other hand, one-stage algorithms, with prominent models like YOLO [12] and SSD [13], directly generate several candidate regions in the input image and classify them based on the type and location of the detection object. Due to simplified image processing steps, one-stage object detection algorithms are faster than two-stage algorithms and have become the most widely applied detection algorithm to date. Liu and Wang [14] enhanced the YOLOv3 model using image pyramid techniques, achieving a 92.39% detection accuracy rate in identifying 12 types of tomato pests, an improvement of 4% over the original model. However, the computational cost required for this model's improvement method is relatively high, and its real-time capability needs further enhancement. She et al. [15] integrated the Feature Pyramid Network (FPN) into SSD, capturing resolution information at different feature levels of the image. The improved SSD model recognized rice pests, especially brown planthoppers, with an average accuracy rate increased from 67.6% to 75.8%. The experimental results indicate that this model performs better in detecting large targets but has substantial room for improvement in recognizing small targets. Moreover, the training network is not very perfected, with precision and performance still needing improvement. Liu et al. [16] combined the Graph Convolutional (GC) network and feature fusion techniques to enhance YOLOv3, achieving a 65.69% detection accuracy rate in identifying 24 types of pests, a 4.27% improvement over the original YOLOv3. However, in actual tests, under complex backgrounds, the model exhibited missed detections of pests, indicating less than ideal performance. Zhang et al. [17] used an improved YOLOv3 to detect tobacco beetle pests, employing the K-means++ algorithm, SIoU Loss, and an improved feature pyramid module to enhance the original YOLOv3 model. The improved model achieved a 93.26% precision rate in recognizing tobacco beetles. However, the dataset for this experiment focused only on tobacco beetles, and in actual detection scenarios, interference from other types of pests or non-pest objects may reduce recognition accuracy and generalizability, affecting its application value in broad agricultural pest management.

In the real world, for tobacco pest detection, model selection not only focuses on recognition performance but also considers the model's resource usage and consumption. For instance, when using mobile devices to detect tobacco pests, the key task is to identify targets efficiently and in real time, providing significant convenience for farmers in early pest control during tobacco cultivation. Therefore, the model's memory footprint and computational resources become critical considerations, directly affecting the model's practical usability [18]. Although traditional models like the YOLO and SSD series have high accuracy rates, their larger sizes make them less suitable for deployment on mobile or embedded devices. To facilitate the operation of deep learning models on mobile devices, numerous researchers have introduced a variety of lightweight network architectures. Zhang et al. [19] used the lightweight feature extraction network GhostNet and depthwise separable convolutions to reconstruct the original YOLOv4 model. In tests on a custom apple dataset, the improved YOLOv4's parameter count was approximately 15.53% of the original model, while the mAP increased by 3.45%. Zhang et al. [20] used MobileNet v2, depthwise separable convolutions, and the Attention Feature Fusion Module (AFFM) module to improve the original YOLOv4 model. Although the accuracy of the improved model decreased slightly, its parameters were reduced to 19.53% of the original YOLOv4. Sun et al. [21] proposed the YOLOv5-CS network based on YOLOv5, C3-light modules, and the SimAM attention module. This model reduced floating-point operations by 15.56% compared to the original YOLOv5, and the average precision (AP) reached 99.1%. Kang et al. [22] used multi-scale features and attention mechanisms to lightweight improve the original CenterNet model, reducing the model size by 25.7% and making it suitable for deployment on actual hardware.

Currently, in natural environments, tobacco pest image detection faces challenges such as similar colors between pests and backgrounds, significant variations in pest target sizes, and the presence of interfering objects in images. This necessitates enhanced model feature extraction capabilities and strengthened focus on target features. Moreover, current pest image detection rarely involves the identification of pest targets in complex growing environments, and the computational cost of the target detection network for tobacco pests is relatively high, necessitating further improvement in its real-time capability. Given the characteristics of tobacco pest detection and the existing issues with pest identification models, this study develops a lightweight model for the identification and monitoring of common tobacco pests in complex environments across different size scales. This paper employs the lightweight AFPN network and VoV-GSCSP module to improve and optimize the original YOLOv8 module, reducing the parameter count and computational load of the original model and incorporating the SimAM attention mechanism to enhance feature extraction capabilities in complex backgrounds and recognition accuracy of pest targets. Through network structure optimization and algorithm design, this study aims to develop a high-performance and lightweight tobacco pest detection algorithm to meet the demands of real-world applications.

2. Materials and Methods

2.1. Image Dataset

This study focused on four common tobacco pests: cutworms, red spiders, aphids, and leafrollers. The dataset was sourced from IP102 [23], a large dataset for pest recognition. Through manual screening and cleaning, 460 pest images were obtained, distributed as follows: 143 for cutworms, 102 for red spiders, 113 for aphids, and 102 for leafrollers. Figure 1 displays samples of pests.

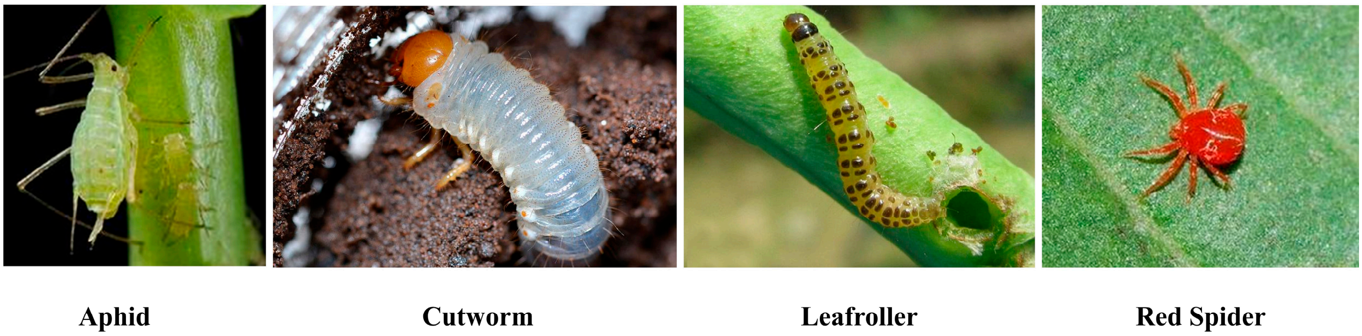


Figure 1. Pest samples.

For data annotation, the study employed the Labelme software (version 5.4.1) for manual labeling [24]. The minimum bounding rectangle of each image was used as the basic information for annotation. Annotations were saved in the YOLO format as txt files.

2.2. Data Enhancement

Data augmentation is a pivotal technique employed in computer vision, serving to enhance the size and diversity of training data, which in turn facilitates improved generalization performance of the detection model [25]. In this study, the Mosaic data augmentation approach was adopted to bolster the training set, aiming to enhance the robustness of the detection model in pest identification [26]. The fundamental principle behind the Mosaic data augmentation technique is to concatenate four images that have been randomly scaled, cropped, and arranged into a single composite image. This strategy aims to expand the data repository and amplify the model's performance in detecting smaller targets. An exemplary outcome of this method can be observed in Figure 2. Through Mosaic data augmentation, multiple smaller images can be amalgamated into a larger composite, thereby enhancing the scale and richness of the dataset without necessitating additional labeled data. To meet deep learning model training requirements, the dataset was expanded to 4000 images through data augmentation, then divided into training and test sets at an 8:2 ratio.

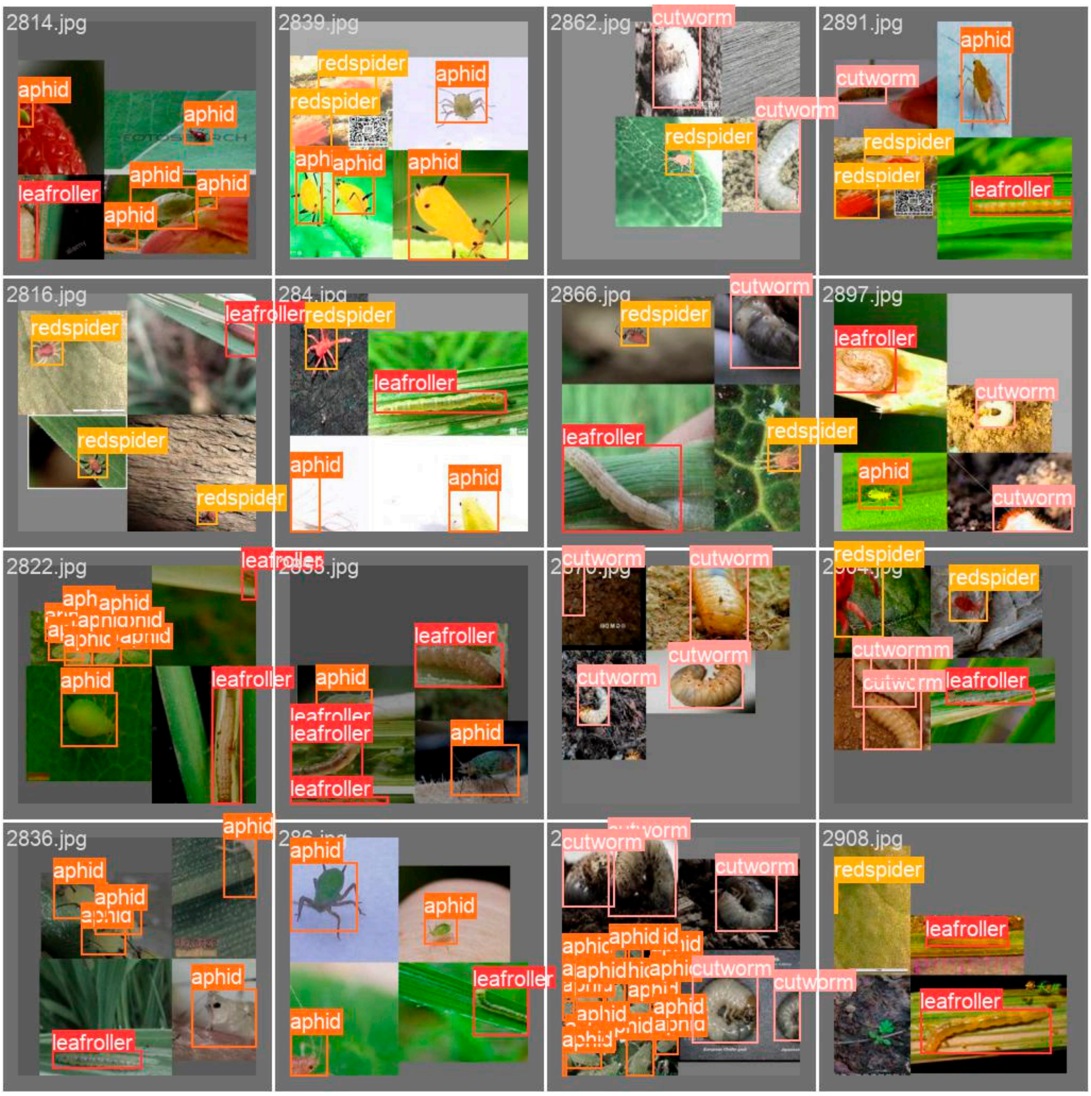


Figure 2. Results after Mosaic data augmentation.

2.3. Standard YOLOv8 Model

YOLOv8 represents the latest installment of the single-stage object detection YOLO series [27,28]. Embracing the detection philosophy of its YOLO predecessors, its modus operandi involves subdividing an image into smaller grid cells, then predicting the center of each grid cell to detect objects [29]. YOLOv8 comprises three main components: a Backbone network, a Neck network, and a Prediction layer. Within its Backbone, the model integrates the Darknet-53 architecture to enhance the feature extraction process, elevating feature extraction quality while curtailing computational overheads. The architecture known as Darknet-53, which comprises fifty-three convolutional layers, is partitioned into several smaller convolutional modules based on varying phases of information transmission, directing the model's gradient flow during propagation, thus mitigating the vanishing gradient issue [30]. YOLOv8's Neck utilizes a Path Aggregation Network-Feature Pyramid Network (PAN-FPN) network structure to construct a feature pyramid, adeptly amalgamating feature maps from the Backbone through various upsampling stages to fuse extracted features [31]. The Prediction or "Head" layer executes object detection predictions and outputs. Here, YOLOv8 introduces a decoupled design, segmenting classification and detection tasks, and assigning them to different branches, augmenting detection efficiency [32]. Its architectural schematic is depicted in Figure 3.

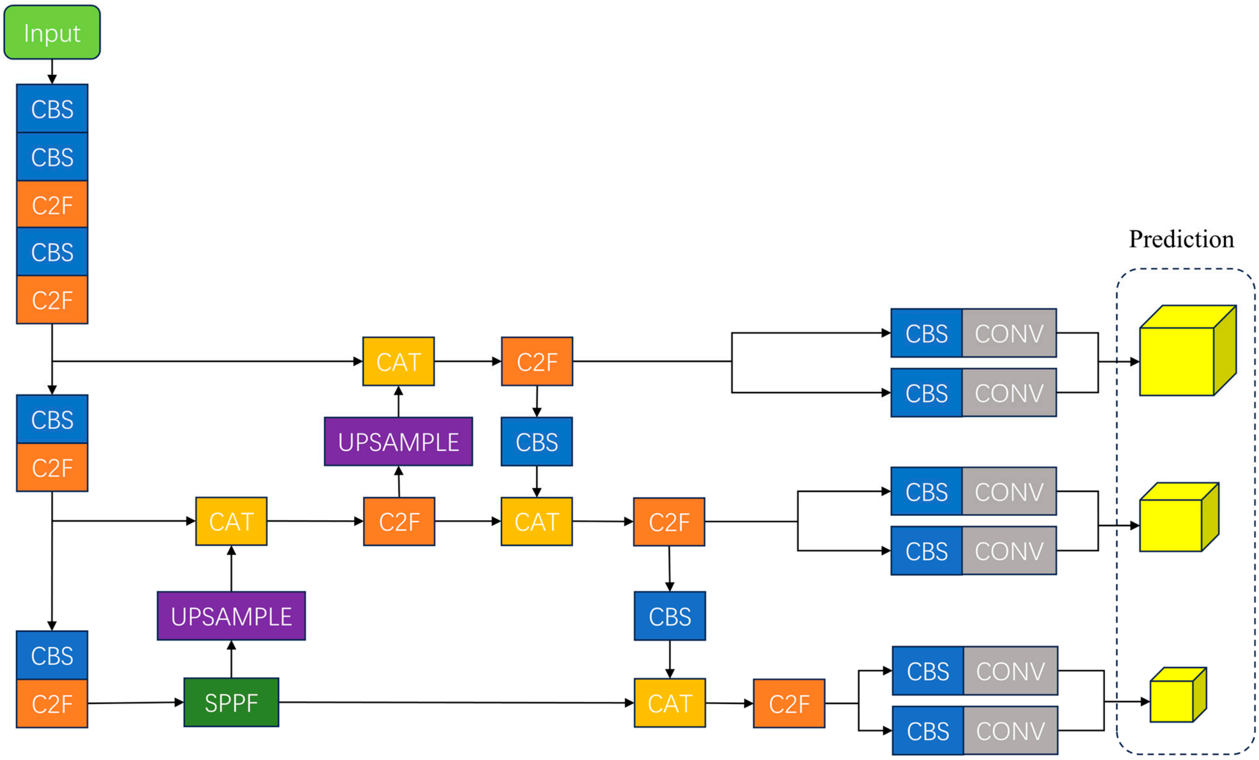


Figure 3. Structure of YOLOv8 model.

2.4. Improved YOLOv8 Model

While the YOLOv8 model exhibits commendable detection accuracy, its extensive parameter count impedes its seamless deployment in detection devices. Hence, there is an exigency for a lightweight design adaptation of YOLOv8. Additionally, in practical pest detection scenarios, challenges abound due to the intricate environments that pests inhabit. These include varying object scales, intricate backgrounds, and instances where detection targets obscure one another. Such complexities pose significant hurdles to the detection task. Thus, judicious refinements to YOLOv8 can bolster both the model's detection performance and efficiency, fostering its deployability in hardware systems.

2.4.1. YOLOv8 Model Improvement Strategy

In this research, the 'l' variant of the YOLOv8 model, designated as YOLOv8l, served as the core framework. To reduce the consumption of computational resources, the final two C2f components within the YOLOv8l structure were substituted by the more efficient VoV-GSCSP module, which is based on the GSConv convolution technique. During the feature fusion phase, a lightweight AFPN network supplanted the original PAN-FPN in YOLOv8l, aiming to alleviate the parameter count of the initial model. The backbone network of YOLOv8l yields four outputs: C1, C2, C3, and C4. These serve as inputs for the AFPN layers. Given the intrinsic challenges associated with pest imagery, such as intricate backgrounds and significant luminance variations due to differing times and weather conditions, the SimAM attention mechanism was integrated into the main network. This enhancement accentuates the focus on pest targets set against complex backdrops. By embedding this attention mechanism within the C2f module of YOLOv8l, higher weightage is assigned to the semantic information of the pest targets, thereby refining detection precision. Figure 4 depicts the structure of the revised network.

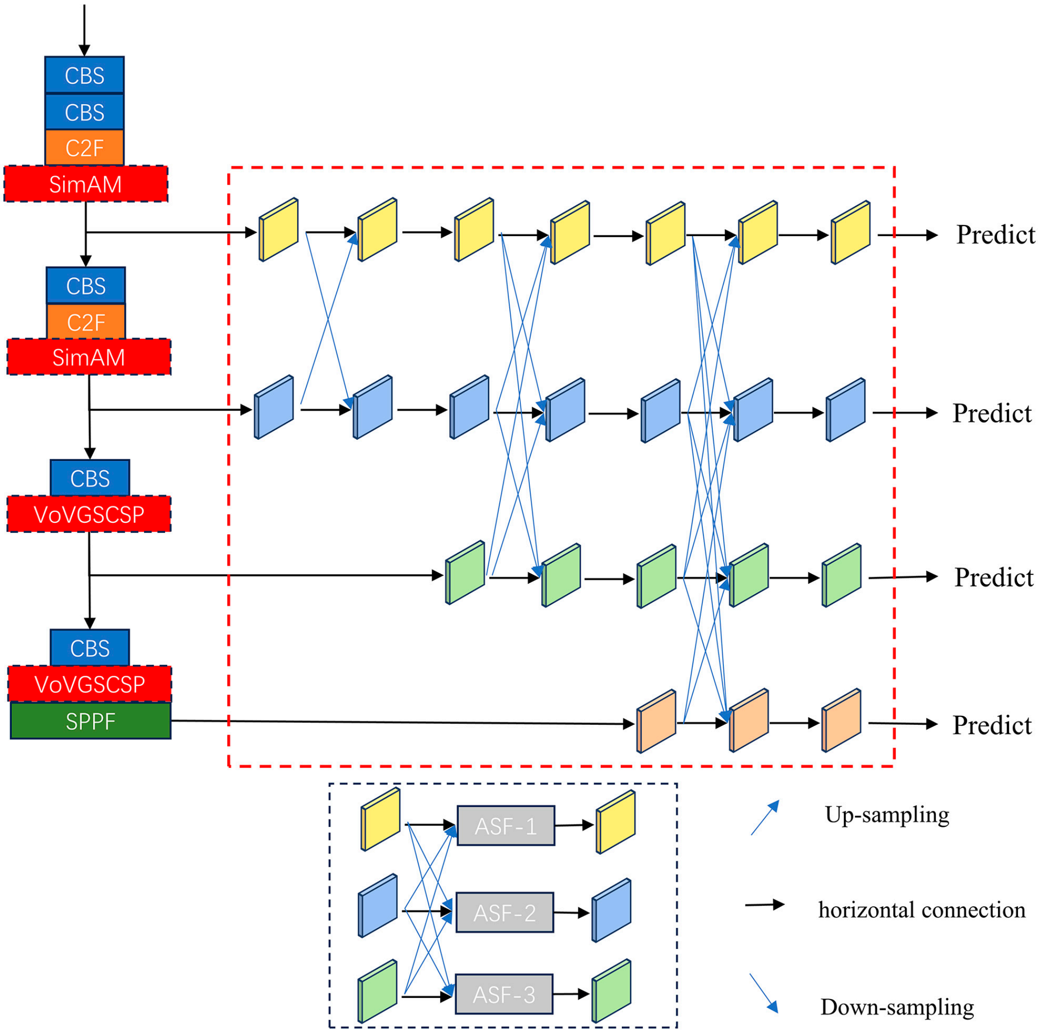


Figure 4. Structure of improved YOLOv8l model.

2.4.2. Asymptotic Feature Pyramid Network

To reduce the parameter count required for network deployment, this study employs the AFPN as a substitute for the PAN-FPN network in the 'neck' layer of YOLOv8 [33]. In the original YOLO algorithm's feature fusion procedure, the FPN and PAN are adopted. This architecture necessitates the fusion and transfer of features across multiple scales, escalating computational overhead [34]. Moreover, the PAN-FPN structure within the YOLO algorithm directly uses the outputs of the C3, C4, and C5 layers from the Backbone during feature fusion, neglecting the semantic disparities among distinct feature layers. Consequently, this results in suboptimal fusion outcomes for non-adjacent layer features [33].

The introduction of the AFPN addresses this issue. AFPN, a feature fusion network, is specifically designed to tackle the substantial semantic fusion gaps between adjacent layers of the Backbone. The architecture of AFPN is asymptotic, harmonizing the semantic information of diverse level features in the progressive fusion process, thus alleviating the aforementioned problem. To maintain uniform dimensions and set the stage for integrating features, the AFPN utilizes 1 × 1 convolution alongside bilinear interpolation to upscale features. During downsampling, various convolutional operations are applied depending on the sampling frequency. Figure 5 displays the configuration of the AFPN network.

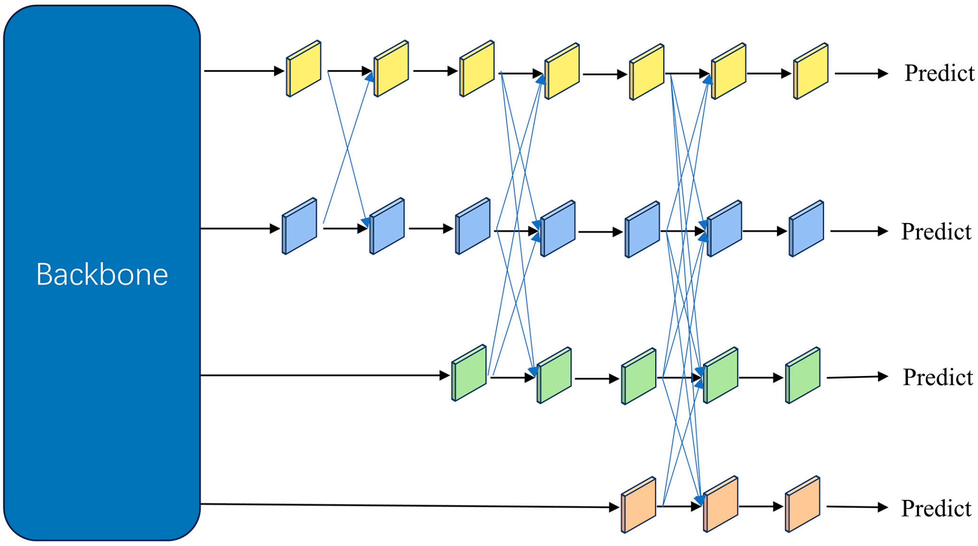


Figure 5. Structure of AFPN network.

The AFPN network introduces a novel feature fusion approach, with a progressive feature fusion strategy that ensures the model's efficiency and accuracy. By substituting the traditional PAN-FPN structure in YOLOv8 with AFPN, this study addresses the semantic gap issues present during the feature fusion process. The progressive architecture of AFPN narrows the semantic disparities between different hierarchical features during fusion, alleviating the problems caused by semantic gaps and thereby enhancing the model's capability in pest detection within complex environments. Furthermore, due to its lightweight design, AFPN offers a more streamlined structure compared to the traditional PAN-FPN. Its integration into YOLOv8 helps to reduce the model's parameter count and lower the overall computational resource footprint.

2.4.3. VoV-GSCSP Module

The VoV-GSCSP module is formulated from the GSConv convolutional framework and the GS bottleneck component, constituting a cross-level part network (GSCSP) module [35]. GSConv, noted for its efficiency, is detailed in Figure 6. By integrating depthwise separable convolution with channel shuffle techniques, GSConv bolsters the module's nonlinear representational capacity, concomitantly mitigating computational overhead.

The GS bottleneck is a module predicated on GSConv, optimizing the conventional bottleneck module. The traditional bottleneck module, originating from ResNet [36], encompasses two standard convolutions: kernels of size 3 × 3 and 1 × 1. The 1 × 1 convolution is primarily employed to diminish and restore the dimensionality of features, while the 3 × 3 convolution establishes the bottleneck layer, characterized by reduced dimensions for both input and output. The distinctive design of the bottleneck effectively manages the dimensions of features, thereby simplifying computational complexity. Figure 7 illustrates the configuration of the GS bottleneck.

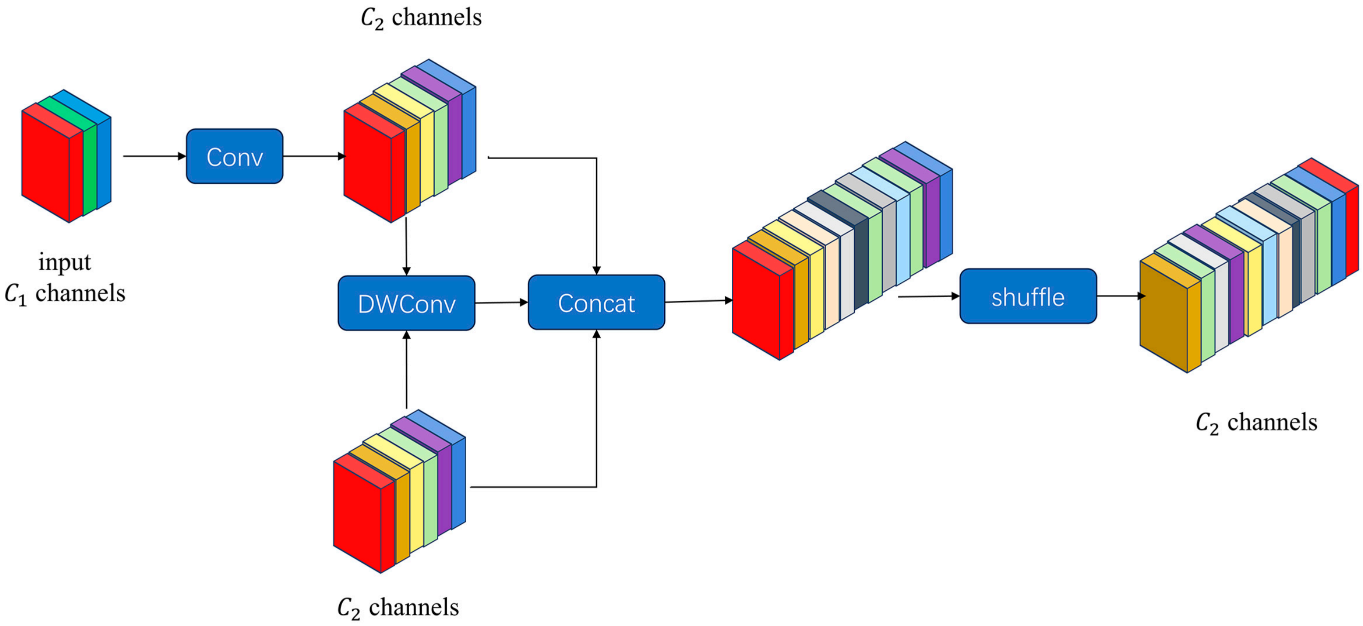


Figure 6. Structure of GSConv module.

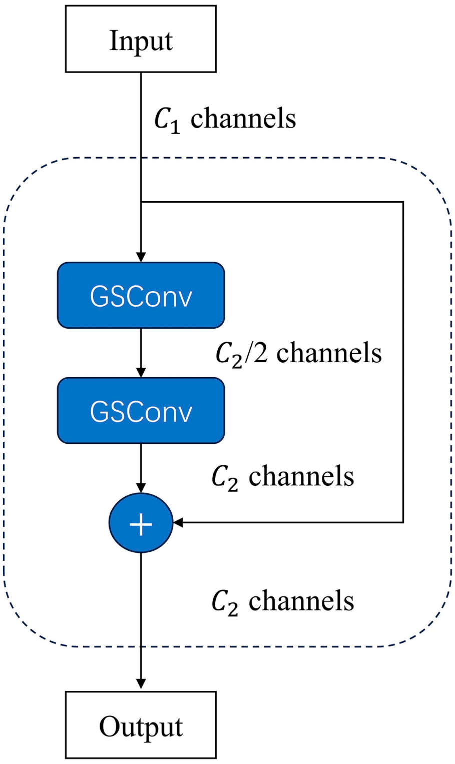


Figure 7. Structure of GS bottleneck.

The VoV-GSCSP represents an ongoing innovation based on the GS bottleneck module, incorporating a single-shot aggregation approach for its design. This module effectively balances the model's accuracy and computational speed, simplifies the network structure, and concurrently maintains high precision and extensive feature reusability. Within VoV-GSCSP, the channel count is divided into two sections: the first section undergoes a Conv convolution initially, then utilizes stacked GS bottleneck modules for feature extraction; the second section serves as a residual connection and simply passes through a single Conv convolution. Ultimately, these two sections are fused based on the channel count and output via a Conv convolution. The structure of VoV-GSCSP is presented in Figure 8.

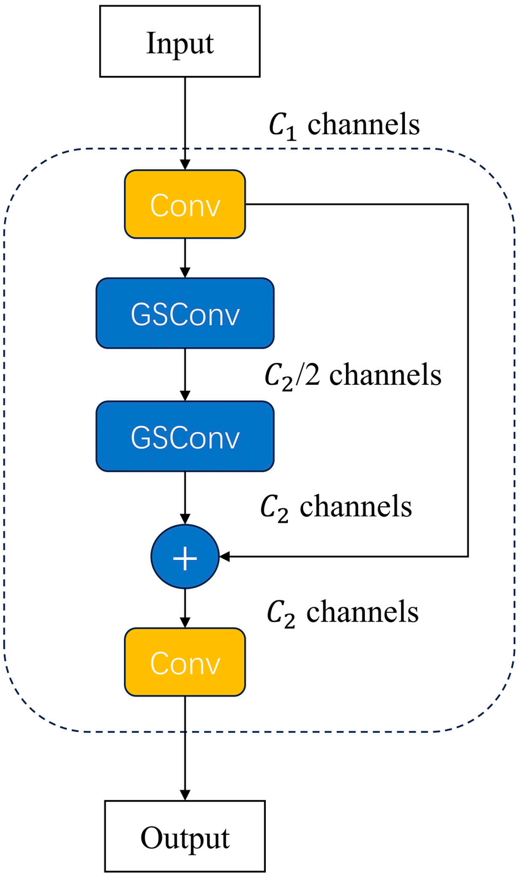


Figure 8. Structure of the VoV-GSCSP module.

The VoV-GSCSP not only retains the advantages of GSConv but also integrates the full suite of benefits from the GS bottleneck. Its innovative skip-connection branch design affords the VoV-GSCSP a more robust nonlinear representation than the C2f module of YOLOv8, effectively overcoming the problem of vanishing gradients. Additionally, it adopts the gradient partitioning strategy from the Cross Stage Partial Network (CSPNet), whereby its channel-splitting technique enhances gradient fusion and reduces redundancy in gradient information, thus fortifying its learning capability. This paper employed the VoV-GSCSP module to replace the final two C2f modules in the YOLOv8l backbone network, aimed at diminishing the computational resource requirements of the model. This allows for efficient operation even in environments with limited computational resources. The lightweight design of the VoV-GSCSP module not only reduces computational complexity but also maintains substantial feature extraction capacity, which is particularly crucial for real-time detection and deployment on edge devices.

2.4.4. SimAM Attention Module

In natural scenes, the recognition of pest images can be challenging due to the similarity between the target and the background, overlap among targets, or the small size of the target. To accurately distinguish between target and non-target information and minimize interference from non-targets, this study integrates an attention mechanism within the primary structure of the network, thus improving its ability to extract features.

Inspired by the human cognitive system, the attention mechanism emulates the human capability to concentrate on particular details, magnifying details of the observed object, and focusing more on the core issues of the data. Within the realm of deep learning, introducing attention mechanisms has been proven to enhance task performance [37]. SimAM, a 3D attention module designed by Yang et al. based on neuroscientific theory [38], differs from traditional channel and spatial attention models. It deduces attention weights based on the energy function of neuroscientific theory, enabling the calculation of neuron significance for feature maps sans extra parameters. The minimum energy function e_i^* for the *i*th neuron in the SimAM attention mechanism is shown in Equation (1).

$$e_t^* = \left(4\left(\lambda + \sigma^2\right)\right) / \left((t-u)^2 + 2\sigma^2 + 2\lambda\right) \quad (1)$$

where

$$u_t = \frac{1}{M-1}\sum_{i=1}^{M-1} x_i, \sigma_t^2 = \frac{1}{M-1}\sum_{i=1}^{M-1}(x_i - u_t)^2 \quad (2)$$

$$M = H \times W \quad (3)$$

Here, u_t represents the average value of all neurons, σ_t^2 is the variance of all neurons, t is the target neuron, x_i is associated with the ith neuron in the input feature map across a singular channel, λ is the regularization coefficient, and M denotes the number of neurons per channel. A smaller e_t^* value indicates that the target neuron in the current feature map is more separable from other neurons, making that neuron more important. The weight for each neuron within the feature map is determined through $1/e_t^*$. Equation (4) displays the ultimate feature map X, where E encapsulates the e_t^* values for all feature map neurons.

$$X = X \cdot \text{sigmoid}(1/E) \quad (4)$$

Incorporating this module into the network significantly enhances its ability to extract essential features, effectively suppressing the interference of non-significant elements. The complex and variable backgrounds in pest images, especially under differing lighting conditions due to changes in time and weather, may pose challenges to pest recognition. The SimAM attention mechanism, by assigning higher weight to the semantic information of pest targets, allows the model to focus more on the pests themselves rather than the surrounding environment. Integrating the SimAM attention mechanism with the feature extraction layers boosts the original model's accuracy in pest detection, particularly when the target pests are small in scale or similar in color to the background. Additionally, most operations of this attention mechanism rely on optimized energy function choices, which avoid excessive structural adjustments. This optimization accelerates the computation of attention weights while maintaining the network's lightness. The structure of the SimAM module is depicted in Figure 9.

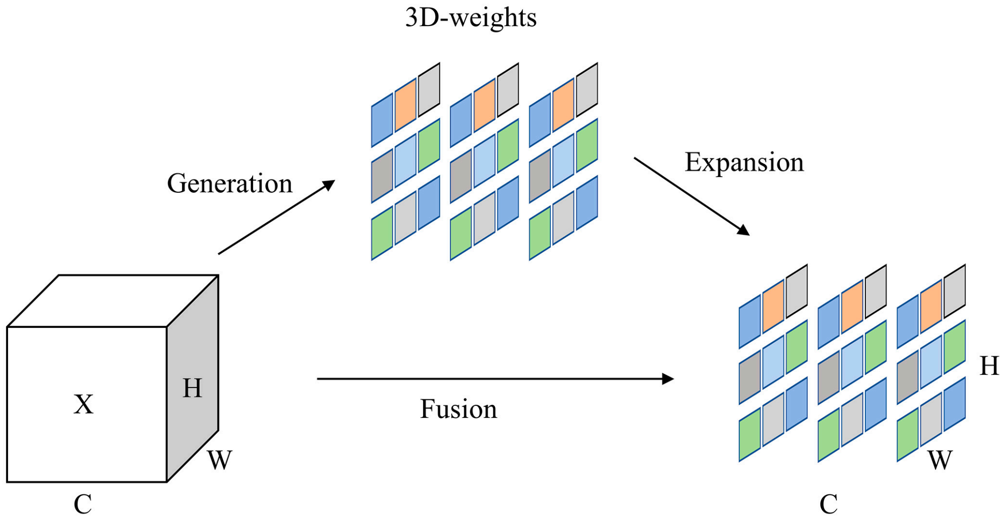


Figure 9. The structure of SimAM attention block, where X is the input feature tensor. C: Channels of the image. W: Width of the image. H: Height of the image.

2.5. Model Evaluation Metrics

The study chose the following as evaluation metrics: Recall, Precision, mAP@0.5, mAP@0.5:0.95, GFLOPs, and Parameters. mAP@0.5 is the mean of the mAP values at an Intersection over Union (IoU) threshold of 0.5, and mAP@0.5:0.95 calculates the mean of mAP values across IoU thresholds from 0.5 to 0.95, increasing by 0.05 at each step. Formulas (5) through (10) are as follows.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$AP = \int_{0}^{1} PRdR \tag{7}$$

$$mAP = \frac{1}{M} \sum\nolimits_{T=1}^{M} AP(T) \times 100\% \tag{8}$$

$$\text{GFlops} = O\left(\sum_{i=1}^{n}, K_i^2 * C_{i-1}^2 * C_i + \sum_{i=1}^{n} M^2 * C_i \right) \tag{9}$$

$$\text{Parameters} = O\left(\sum_{i=1}^{n}, M_i^2 * K_i^2 * C_{i-1} * C_i \right) \tag{10}$$

In Equations (5) and (6), *TP* refers to true positives, *FN* to false negatives (missed detections), *TN* to true negatives, and *FP* to false positives (false detections). In Equation (8), *M* represents the total number of classes involved in the detection task, and *AP(T)* denotes the detection precision of class *T*. In Equations (9) and (10), *O* represents the order of magnitude, *K* is the kernel size, *C* is the channel number, *M* is the input image size, and *i* is the iteration number.

3. Results

3.1. Experiment Settings

3.1.1. Experimental Platform

The experimental setup's hardware utilizes an Intel(R) Xeon(R) Gold 6240 CPU, featuring 24 cores and 48 threads (base frequency at 2.60 GHz). For GPU acceleration, two NVIDIA GeForce RTX 3090 units are used, leveraging the CUDA 11.6 platform to expedite the network training process. The training environment operates on Python 3.8.16, with the PyTorch 1.13.1 deep learning framework, on an Ubuntu OS (version 22.04.3) Development tasks are conducted using the PyCharm IDE (version 2023.1.2).

3.1.2. Model Training Strategy

In this research, training and testing data were segmented into various groups (Batch size), and after a comparative study, it was decided to train with 32 images per batch. An Epoch, meaning one cycle through all images in the dataset, was observed to reach a point of convergence in the network's loss value after 100 cycles. Thus, the experiment was designed with a 100 Epoch completion target.

Before processing, all images underwent resizing to a standard dimension of 224 × 224 pixels. To preserve the integrity of the images, no data augmentation techniques were applied during the training phase. The learning rate adjustments were managed by the Stochastic Gradient Descent (SGD) optimizer, starting at 0.01, with a momentum factor of 0.937 and a weight decay factor of 0.0005. For evaluating the model's accuracy on the test dataset, an IoU benchmark of 0.7 was used, alongside non-maximum suppression (NMS) with a matching IoU criterion of 0.7. Furthermore, to maintain unbiased model assessment, no pre-trained models were leveraged in any comparative or ablation analysis.

3.2. Comparison Experiment before and after Model Improvement

This study contrasts the training outcomes of the original YOLOv8l and the improved YOLOv8l models. The experiments constitute a univariate test, with both models being trained and tested under an identical platform environment. Upon completion of the training phase, a comparative evaluation of both models' efficacy on the test dataset was undertaken to determine the impact of the modifications. Table 1 displays the outcomes of this evaluation.

Table 1. Performance of YOLOv8l model before and after improvement.

Model	Parameters	GFLOPs	mAP@0.5	Recall	mAP@0.5:0.95	Precision
YOLOv8l	43.61 M	164.8	87.9%	77.4%	69.1%	90.3%
Improved-YOLOv8l	20.65 M	131.9	88.9%	80.1%	69.7%	92.7%

Table 1 data reveals that the enhanced YOLOv8l model has made notable progress in terms of compactness. The metrics for the revised YOLOv8l stand at 20.65M parameters and 131.9 GFLOPs, marking reductions of 52.66% and 19.9%, respectively, when juxtaposed with the original YOLOv8l version. Regarding accuracy in detection, the enhanced YOLOv8l version consistently surpassed the original across the board in the test dataset. In detail, mAP@0.5 saw an uplift of 1%, Recall by 2.7%, mAP@0.5:0.95 by 0.6%, and Precision by 2.4%.

For a graphical representation and evaluation of both algorithms' efficacy in identifying pests in tobacco crops, a Precision–Recall (P-R) graph was constructed, positioning Recall on the *x*-axis against Precision on the *y*-axis. An IoU threshold of 0.5 was employed to segregate true positives from false positives. This P-R graph is shown in Figure 10.

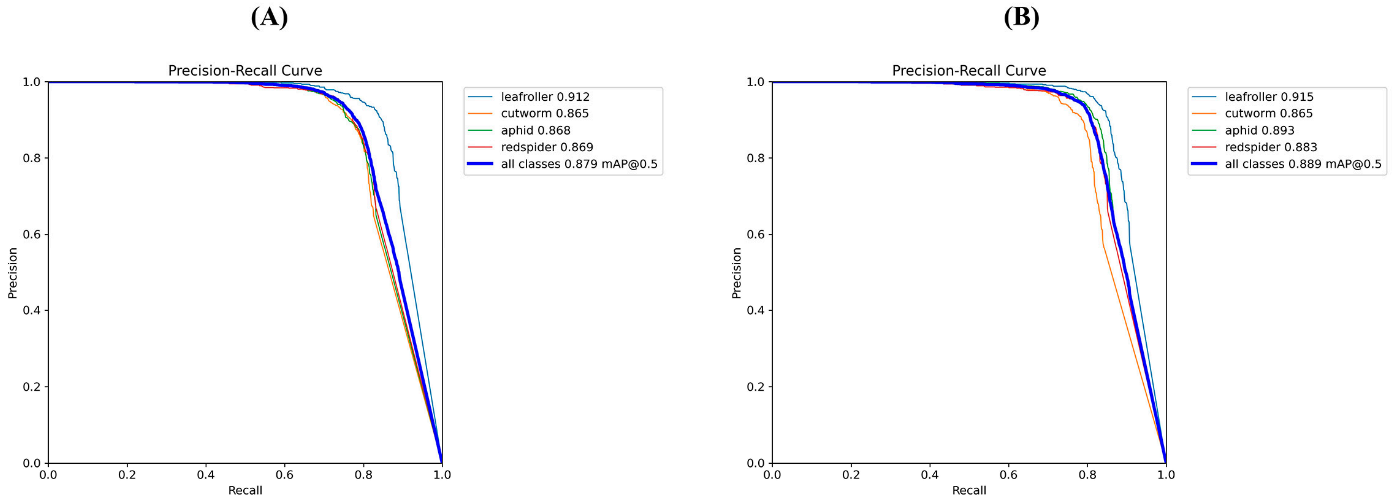


Figure 10. P-R curve: (**A**) YOLOv8l, (**B**) Improved-YOLOv8l.

From the graph, it is evident that the mAP@0.5 score of the refined model reached 88.9% for detecting four types of tobacco pests. For detection of leafrollers, cutworms, aphids, and red spiders, the mAP@0.5 values were 91.5%, 86.5%, 89.3%, and 88.9%, respectively, showing improvements of 0.3%, 2.5%, and 1.5% relative to the original YOLOv8l. Combining all the aforementioned data, it can be concluded that the advancements in the YOLOv8l model have not just boosted its detection capabilities but also markedly lessened both the parameter size and computational load, achieving no-table progress in model lightweighting.

3.3. Ablation Experiments

To evaluate the effectiveness of the suggested enhancements on the baseline model, eight ablation studies were performed on the upgraded model utilizing identical hardware and datasets. The groups for the ablation studies included the YOLOv8l equipped with the AFPN network (YOLOv8l+AFPN), the YOLOv8l utilizing the SimAM attention mechanism (YOLOv8l+SimAM), the YOLOv8l incorporating the VoV-GSCSP module (YOLOv8l+VoV-GSCSP), and combinations of various methods: YOLOv8l+AFPN+VoV-GSCSP, YOLOv8l+AFPN+SimAM, and YOLOv8l+VoV-GSCSP+SIMAM. The comparative results of the ablation tests are presented in Table 2.

Table 2. The results of the ablation test.

Model	Parameters	GFLOPs	mAP@0.5
YOLOv8l	43.61 M	164.8	87.9%
YOLOv8l+AFPN	27.67 M	151.6	88.4%
YOLOv8l+SimAM	43.61 M	164.8	88.4%
YOLOv8l+VoV-GSCSP	36.60 M	145.2	87.9%
YOLOv8l+AFPN+VoV-GSCSP	20.65 M	131.9	88.5%
YOLOv8l+AFPN+SimAM	27.67 M	151.6	88.7%
YOLOv8l+VoV-GSCSP+SIMAM	36.60 M	145.2	88.2%
Improved+YOLOv8l	20.65 M	131.9	88.9%

Data from Table 2 indicates that the AFPN network contributed most significantly to both model lightweighting and detection accuracy enhancement. When the AFPN network replaced the feature fusion network of the original YOLOv8l model, there was a 36.55% reduction in the model's parameter count and a 0.5% increase in the mAP@0.5 value. However, the AFPN network's ability to reduce the original model's GFLOPs was limited, achieving only an 8.01% decrease. This is attributed to the 4 × 4 and 8 × 8 convolutions used in the downsampling operations of the AFPN network. Larger kernel convolutions introduce additional computational overhead, resulting in the model's computational complexity not being significantly reduced. The VoV-GSCSP module, built from GSConv, made the most substantial contribution to the decline in the original model's GFLOPs. By combining depthwise separable convolution and channel shuffling methods, GSConv successfully minimizes model complexity without compromising its accuracy. After incorporating the VoV-GSCSP module, the GFLOPs of YOLOv8l decreased by 11.9%. The introduction of the SimAM attention mechanism also enhanced the model's detection precision, achieving a 0.5% improvement over the original model. Given the no-additional-parameter feature of SimAM, the model's parameters and GFLOPs remained unchanged after integrating SimAM. When combining multiple optimization techniques, further improvements in model performance were observed. Combining the AFPN network with the VoV-GSCSP module, the model's parameters and GFLOPs decreased by 52.65% and 19.96%, respectively. Integrating the AFPN network with the SimAM attention mechanism elevated the model's mAP score to 88.7%. Utilizing all three optimization techniques in tandem, the model performance was optimized. The results from the ablation tests reveal that all three proposed optimization strategies contributed positively to the comprehensive performance of the model. The improved model exhibited notable advancements in terms of lightweighting, achieving an mAP@0.5 value of 88.9% with only 20.65 M parameters and 131.9 GFLOPs, thereby confirming the efficiency and practicality of the suggested enhancements.

Additionally, this paper offers a visual analysis of the performance of the YOLOv8l models on the test set, each incorporating one of the three optimization strategies. Precision, Recall, mAP@0.5, and mAP@0.5:0.95 were selected as the criteria for evaluation. The changes in these four indicators following the addition of different modules to YOLOv8l are depicted in Figure 11.

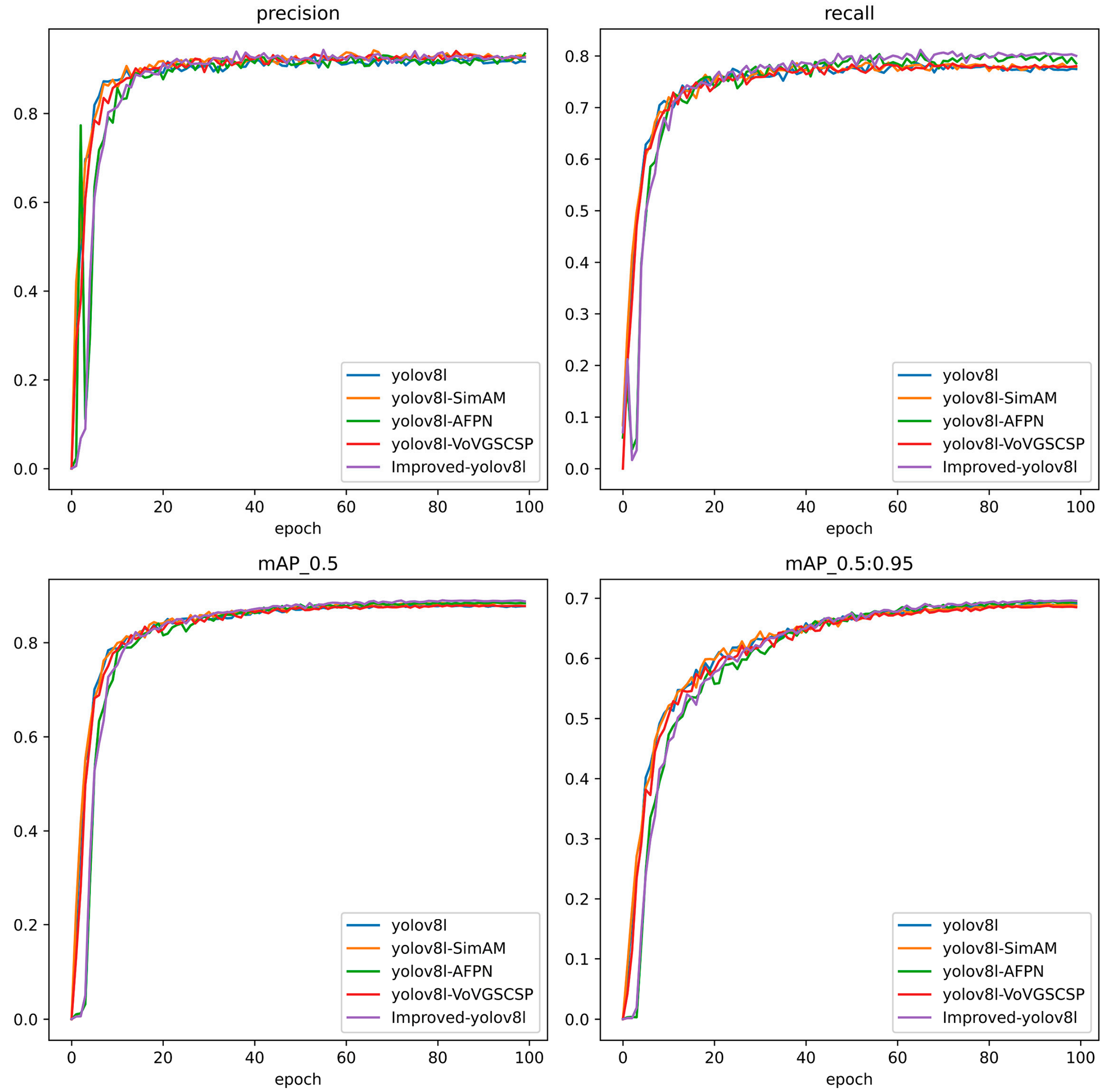


Figure 11. Change curves of four indicators after adding different modules to YOLOv8l.

Referencing Figure 11, it is noticeable that the model's performance on the test set was variably affected by diverse optimization approaches. However, broadly speaking, training curves of all models were similar, showing rapid convergence. This indicates that the lightweighting strategies employed in the study did not compromise the learning capability of the original model.

3.4. Different Model Performances

For a comprehensive evaluation of the model introduced in this research, we juxtaposed its performance against leading-edge object detection frameworks such as YOLOv5 [39], YOLOv6 [40], and YOLOv8. All models are single-stage object detection algorithms. Notably, the versions of YOLOv5 and YOLOv6 utilized were the latest official anchor-free implementations, which demonstrate significant improvements over their original versions. The hardware and software configurations, as well as the hyperparameters used for training all models, were kept consistent. The outcomes of the experiments are displayed in Table 3.

Table 3. Performance comparison of different models.

Model	mAP@0.5	Recall	Precision	mAP@0.5:0.9	Parameters
YOLOv5m	85.5%	74.6%	90.3%	63.5%	25.05 M
YOLOv6m	87.3%	77.4%	93.6%	67.1%	51.98 M
YOLOv8m	86.7%	76.5%	90.6%	65.6%	25.84 M
YOLOv8l	87.9%	77.4%	91.6%	69.1%	43.61 M
Improved-YOLOv8l	88.9%	80.1%	92.7%	69.7%	20.65 M

The data in Table 3 reveals that in terms of accuracy metrics and model parameters, the improved model exhibits superior performance. Compared to YOLOv5m, YOLOv6m, YOLOv8m, and YOLOv8l, the mAP@0.5 value of Improved-YOLOv8l is 3.4%, 1.6%, 2.2%, and 1% higher, respectively. In terms of Recall, the improvements are 5.5%, 2.7%, 3.6%, and 2.7% respectively. Moreover, the model parameters are reduced by 17.6%, 60.3%, 20%, and 52.6% respectively. Collectively, these indicators suggest that the improved YOLOv8l model demonstrates robust detection capabilities in complex environments, achieving high detection accuracy while being lightweight in size, outperforming YOLOv5m, YOLOv6m, YOLOv8m, and YOLOv8l.

Using Precision, Recall, mAP@0.5, and mAP@0.5:0.95 as evaluation metrics, the performance curves of different models on the test set were plotted, as shown in Figure 12.

Figure 12 indicates that the convergence of the curves for the five models is similar, with each model beginning to converge by the 20th epoch. Ultimately, improved YOLOv8l model demonstrates a slight edge. Specifically, for the accuracy metrics mAP@0.5 and mAP@0.5:0.95, both the original YOLOv8l and Improved-YOLOv8l exhibit excellent performance, with YOLOv5m lagging behind and YOLOv6m showing considerable fluctuations. For the recall curve, the final result of Improved-YOLOv8l surpasses other models. For the Precision curve, all five models showcase comparable performances. Through the model performance curve comparisons, it is evident that the improved YOLOv8l model possesses robust capabilities in recognizing positive class samples, effectively capturing target categories comprehensively.

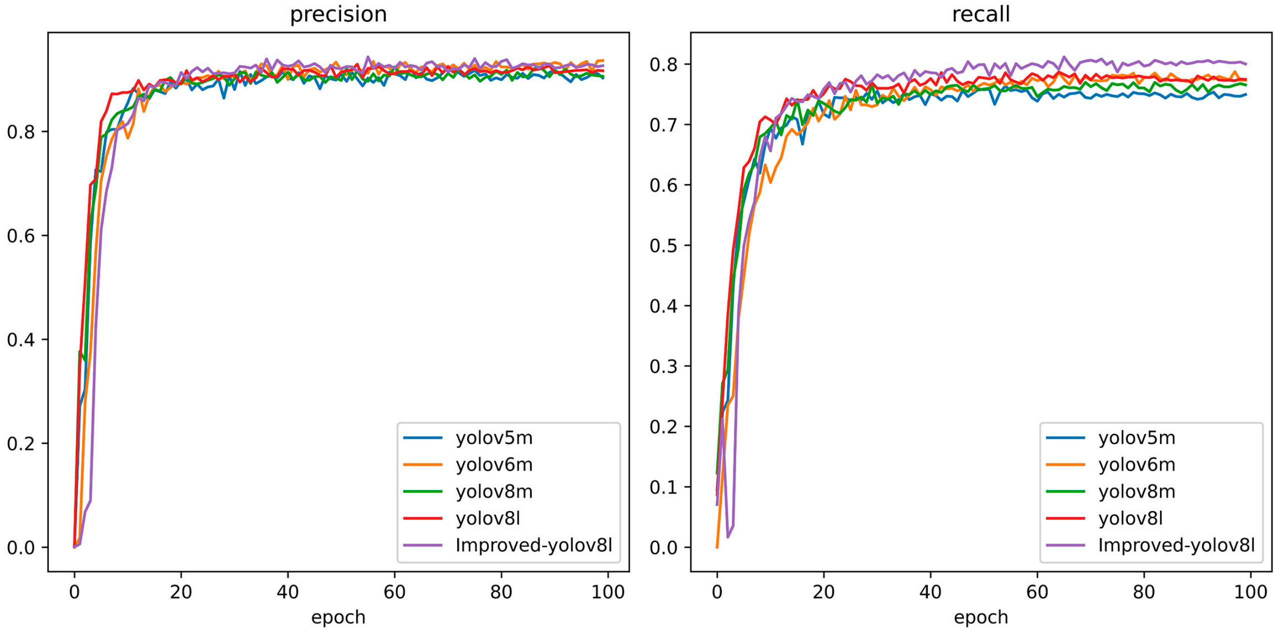


Figure 12. *Cont.*

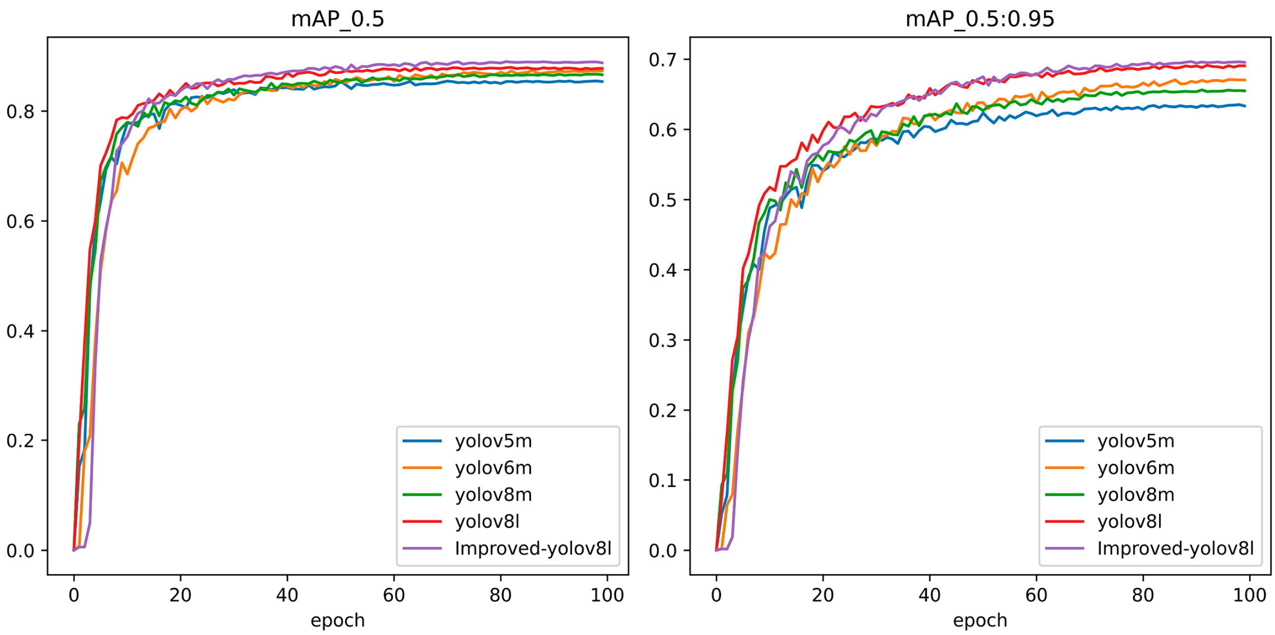
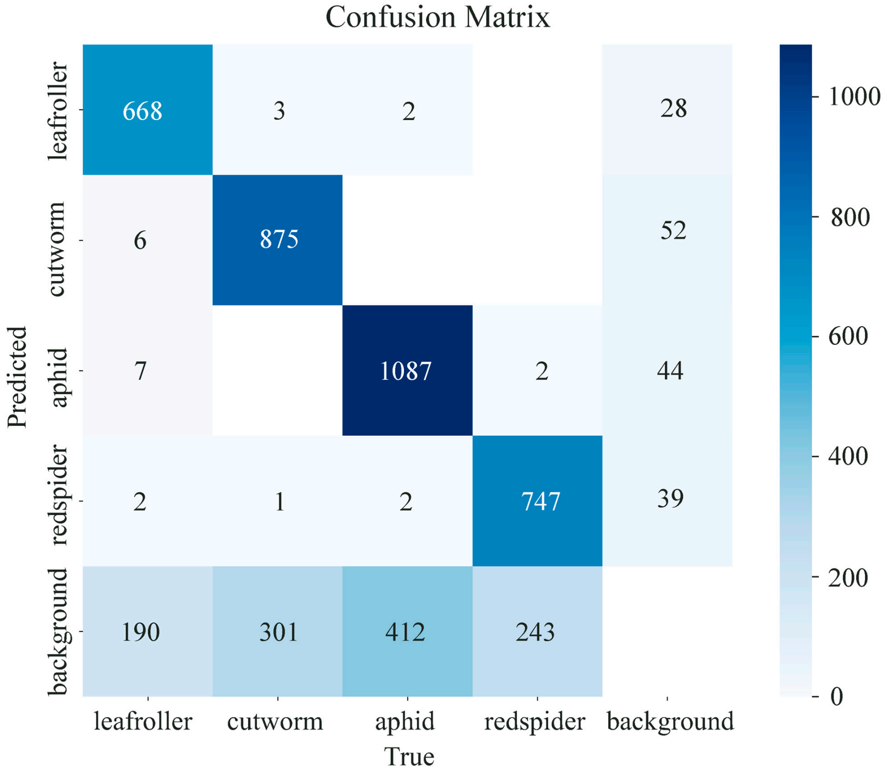


Figure 12. Performance curves of different models on the test set.

3.5. Model Detection Results

To evaluate the model's proficiency in recognizing the four pests of tobacco in complex environments, this study constructed confusion matrices based on the testing dataset outcomes, illustrated in Figures 13 and 14. Within this confusion matrix, the maximum predicted classifications consistently lie along the diagonal, thereby confirming the feasibility of the model for pest detection. The recall rates for the detection of the four pests derived from the confusion matrix are detailed in Table 4.



	leafroller	cutworm	aphid	redspider	background
leafroller	668	3	2		28
cutworm	6	875			52
aphid	7		1087	2	44
redspider	2	1	2	747	39
background	190	301	412	243	

Figure 13. Confusion matrix: YOLOv8l.

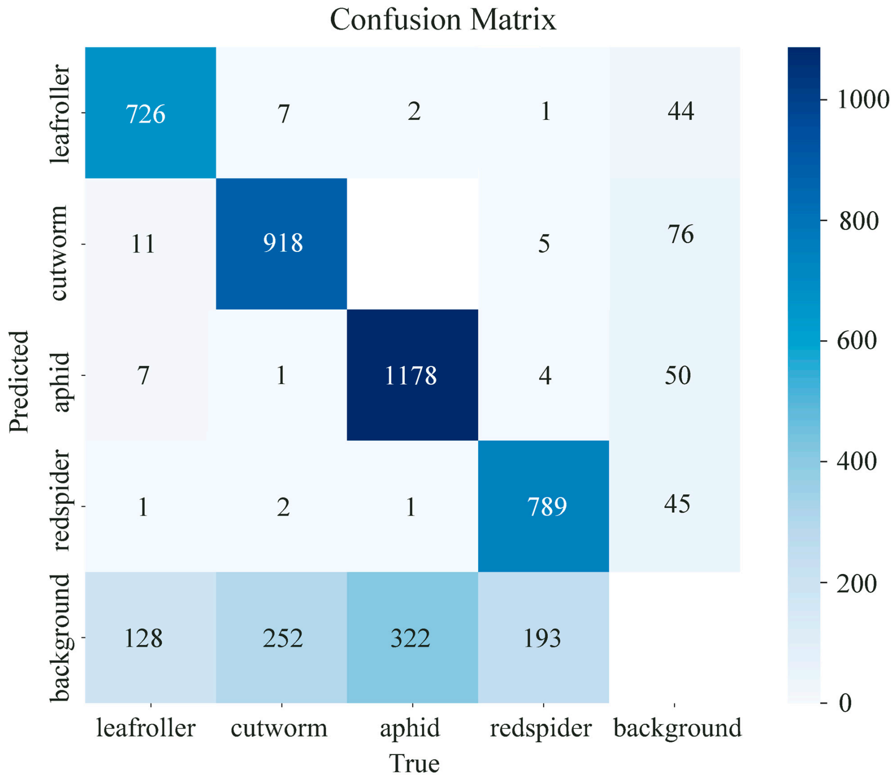


Figure 14. Confusion matrix: Improved-YOLOv8l.

Table 4. Recall results for four pests.

Model	Leaf Roller	Cutworm	Aphid	Red Spider
YOLOv8l	80.6%	76.1%	75.4%	77.2%
Improved-YOLOv8l	83.6%	78.2%	79%	79.5%

The data presented in the table clearly demonstrates that the enhanced YOLOv8l model surpasses the original YOLOv8l in recall rate for all categories. The improved model boasts recall rates of 83.6% for leafrollers, 78.2% for cutworms, 79.0% for aphids, and 79.5% for red spiders, marking increases of 3.0%, 2.1%, 3.6%, and 2.3%, respectively, over the original YOLOv8l. The increase is particularly noteworthy for aphids, with a significant 3.6% improvement in recall rate. As illustrated in Figure 13, the original model is prone to misclassification when pests exhibit color and size characteristics similar to the background. For instance, leafrollers can sometimes be mistaken for the background due to their camouflaging properties. Cutworms may occasionally be confused with leafrollers, likely due to their overlapping morphological features. Such misclassifications, manifesting as false negatives, are especially prevalent with aphids, whose small size renders them less conspicuous against various backgrounds, leading the original model to confuse them with the background. The improved YOLOv8l model, with its integrated attention mechanism module, has improved feature extraction capabilities, reducing such errors. This is evident from the decreased number of false positives and false negatives in the confusion matrix for aphids and other pests, as can be seen in Figure 14. The reduction in false positives and negatives contributes to the enhanced recall rates for all pests, as detailed in Table 4, with the improvements in the aphid category being the most pronounced. These enhancements confirm that the model has successfully addressed some of the challenges associated with pest detection in complex backgrounds, thereby affirming an increase in the model's precision in distinguishing pests from the background.

To showcase the detection results of the enhanced YOLOv8l model, random images were selected from the test dataset for comparison. These findings are illustrated in Figure 15. Highlighted areas represent the network's detection results, with the text on top of the boxes indicating the recognized pest type and numerical values reflecting the detection confidence.

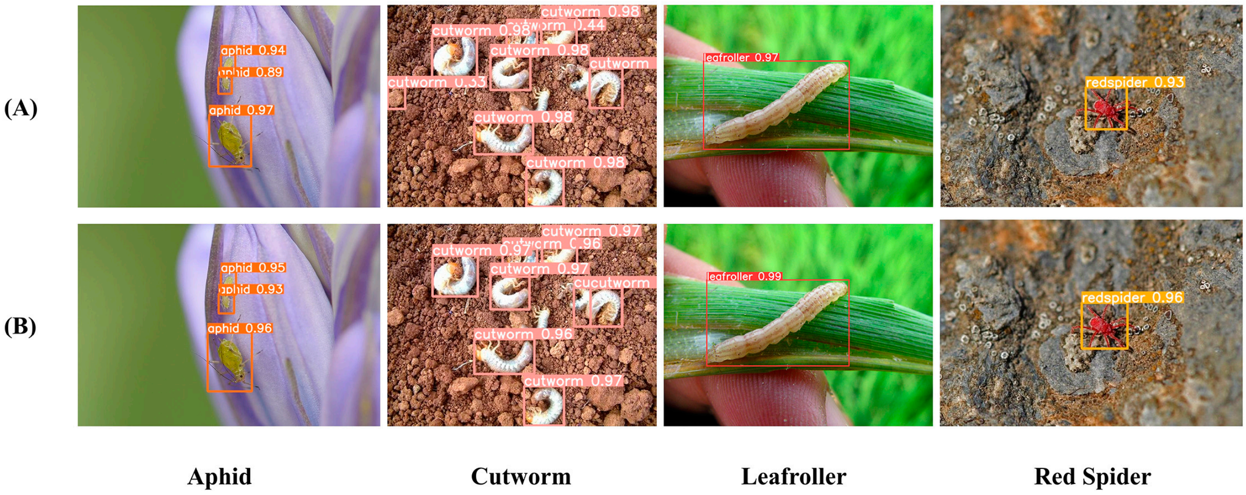


Figure 15. Contrasting the detection outcomes pre and post enhancement of the model: (**A**) YOLOv8l, (**B**) Improved-YOLOv8l.

From Figure 15, it is evident that both models exhibit comparable detection performance. However, in certain scenarios, the improved YOLOv8l model excels. In tests recognizing smaller targets like aphids and red spiders, the improved YOLOv8l model registers higher confidence levels. This is attributable to the YOLOv8l model integrating the AFPN network, thereby expanding the receptive field of the original model, enhancing its sensitivity and adaptability towards smaller object detection. In intricate backgrounds, the improved YOLOv8l model similarly holds an advantage. In recognizing cutworms, the original YOLOv8l model displays lowered confidence for overlapping and occluded objects and occasionally misidentifies background elements. In contrast, the improved YOLOv8l model avoids these pitfalls, consistently exhibiting a detection confidence of over 96% for overlapping and occluded objects without false positives. Taking the test results into account, the improved YOLOv8l model demonstrates superior detection capabilities and confidence in recognizing small targets and intricate backgrounds compared to the original YOLOv8l. Given the more lightweight nature of the improved YOLOv8l model, it holds a distinct advantage in practical scenarios.

3.6. Model Interpretability Analysis

To evaluate the improved YOLOv8l model's ability to extract pest features via its backbone network, this paper employs the Grad-CAM (Gradient-weighted Class Activation Mapping) algorithm for visual analysis [41]. Grad-CAM generates a heatmap by calculating the gradient of the input image, highlighting the key regions that the model primarily focuses on. Brighter colors indicate a higher degree of attention from the model. This paper derives the final layer's output feature map from the backbone network, acquiring the activation pattern through gradient computation. The activation distribution is then superimposed on the original image, creating a heatmap, depicted in Figure 16.

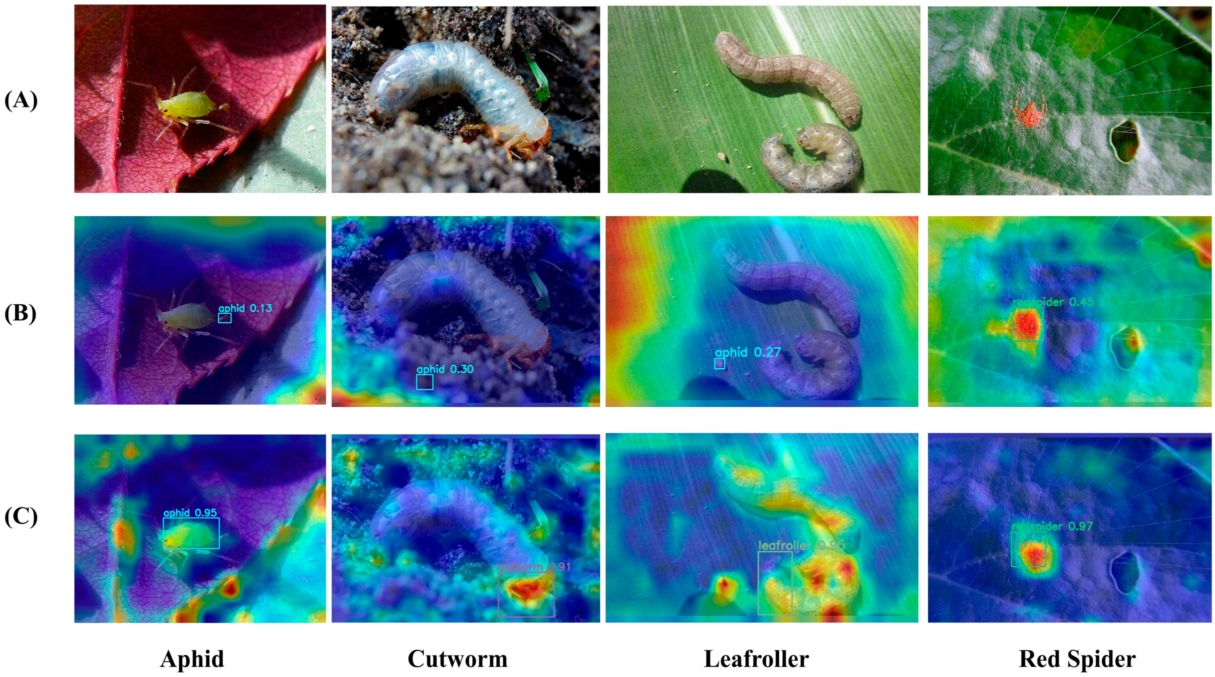


Figure 16. Comparison of model heat maps: (**A**) Original images, (**B**) YOLOv8l, (**C**) Improved-YOLOv8l.

The visualization outcomes reveal that the enhanced YOLOv8l model shows notable improvements in feature extraction over its predecessor. Prior to the introduction of the SimAM attention mechanism and the VoV-GSCSP module, the original model demonstrated a lower degree of focus on pest targets, as reflected in the heatmap with more blue areas, indicating a lower level of confidence and suggesting that the original model's backbone network had limited capability in extracting effective pest features. However, with the incorporation of the SimAM attention and VoV-GSCSP, the model's backbone network became more precise in extracting valid information from the image, with its decision focus centered on the pest target area itself. This indicates that the improvement strategies employed in this paper can significantly enhance the model's decision reliability.

4. Conclusions

(1) Based on the characteristics and challenges of recognizing pests of tobacco in complex environments, this study introduces a refined, more efficient YOLOv8 model. Key enhancements include implementing the streamlined AFPN network and VoV-GSCSP modules as substitutes for the original feature extraction network and C2f module of YOLOv8, targeting a reduction in the model's parameter count and processing requirements. Furthermore, it incorporates the SimAM attention module to boost the model's accuracy in feature extraction and pest target localization against complex backdrops.

(2) This study carried out comparative tests using the test dataset, adopting Precision, Recall, Parameters, GFLOPs, mAP@0.5, and mAP@0.5:0.95 as metrics to evaluate the efficacy and detection performance of the enhancement strategies. Findings from the comparative analysis indicate that, in relation to the baseline model, the enhanced YOLOv8 version exhibits a decrease in parameter count and GFLOPs by 52.66% and 19.9% respectively, an augmentation in the mAP@0.5 score by 1%, an enhancement in Recall by 2.7%, an uplift in mAP@0.5:0.95 by 0.6%, and a boost in Precision by 2.4%. When benchmarked against prevalent object detection frameworks such as YOLOv5m, YOLOv6m, and YOLOv8m, the advanced YOLOv8 version showcases superiorities in detection precision and parameter efficiency.

(3) The enhanced YOLOv8 model effectively balances model resource consumption with detection accuracy. It achieves a significant reduction in model parameters and computational requirements while enhancing the model's detection capabilities. This advancement

allows for the model to be readily deployed on resource-constrained embedded devices, such as mobile terminals, facilitating real-time and accurate identification of tobacco pests in complex environments, thus demonstrating considerable practical application value. To actualize the application of this model in agricultural machinery, it can be integrated with lightweight processors and custom software interfaces, enabling seamless interaction with smart plant protection devices or ground robots. Such equipment is capable of processing image data in real-time in the field, rapidly and accurately identifying and locating pests, providing decision support for precision pesticide application.

Author Contributions: Data curation, H.Z. and M.Y.; Formal analysis, H.Z.; Funding acquisition, D.S.; Investigation, K.Z.; Methodology, D.S. and K.Z.; Project administration, D.S., J.X., W.W., and J.L.; Resources, X.X.; Software, H.Z.; Supervision, J.X., W.W., and J.L.; Validation, K.Z. and M.Y.; Visualization, X.X.; Writing—original draft, D.S. and K.Z.; Writing—review and editing, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Special Fund for Construction of Modern Agricultural Industrial Technology System Innovation Team of Guangdong Province under Grant 2023KJ120 and 2023KJ108, the 2023 China Tobacco Corporation key research and development project under Grant 110202301011, the 2023 Guangdong Tobacco Association Research Project under Grant H20230340, and the Guangzhou Science and Technology Plan Project under Grant 202002030245.

Data Availability Statement: The data and code for this study can be accessed in the Tobacco_Pest_YOLOv8 repository, available at https://github.com/escape2019/Tobacco_Pest_YOLOv8, accessed on 1 January 2024.

Acknowledgments: We would like to express our gratitude to all authors for their technical assistance in this study, as well as to the anonymous reviewers for their valuable suggestions and critical comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chen, P.; Zhang, H.; He, W. Multi-scale feature fusion method for bundled tobacco leaf classification based on fine-grained classification network. *J. Anhui Agric. Univ.* **2022**, *49*, 1013–1021. [CrossRef]
2. Qu, Y.; Li, J.; Chen, Z.; Wen, Z.; Liang, J.; Cao, Y.; Li, S.; Chen, J. Current Status and Future Development of Flue-cured Tobacco Production in Guangdong Province. *Guangdong Agric. Sci.* **2019**, *46*, 141–147. [CrossRef]
3. Rabb, R.L.; Todd, F.A.; Ellis, H.C. Tobacco Pest Management. In *Integrated Pest Management*; Apple, J.L., Smith, R.F., Eds.; Springer: Boston, MA, USA, 1976; pp. 71–106. ISBN 978-1-4615-7269-5.
4. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep Learning in Agriculture: A Survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
5. Santos, L.; Santos, F.N.; Oliveira, P.M.; Shinde, P. Deep Learning Applications in Agriculture: A Short Review. In Proceedings of the Robot 2019: Fourth Iberian Robotics Conference, Porto, Portugal, 20–22 November 2019; Springer: Cham, Switzerland, 2020; pp. 139–151.
6. Bian, K.-C.; Yang, H.-J.; Lu, Y.-H.; Lu, Y.-H. Application Review of Deep Learning in Detection and Identification of Agricultural Pests and Diseases. *Softw. Guide* **2021**, *20*, 26–33.
7. Liu, Z.; Gao, J.; Yang, G.; Zhang, H.; He, Y. Localization and Classification of Paddy Field Pests Using a Saliency Map and Deep Convolutional Neural Network. *Sci. Rep.* **2016**, *6*, 20410. [CrossRef] [PubMed]
8. Wang, R.; Zhang, J.; Dong, W.; Yu, J.; Xie, C.; Li, R.; Chen, T.; Chen, H. A Crop Pests Image Classification Algorithm Based on Deep Convolutional Neural Network. *TELKOMNIKA (Telecommun. Comput. Electron. Control.)* **2017**, *15*, 1239–1246. [CrossRef]
9. Cheng, X.; Zhang, Y.; Chen, Y.; Wu, Y.; Yue, Y. Pest Identification via Deep Residual Learning in Complex Background. *Comput. Electron. Agric.* **2017**, *141*, 351–356. [CrossRef]
10. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
11. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
12. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
13. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Volume 9905, pp. 21–37.
14. Liu, J.; Wang, X. Tomato Diseases and Pests Detection Based on Improved Yolo V3 Convolutional Neural Network. *Front. Plant Sci.* **2020**, *11*, 898. [CrossRef]

15. She, H.; Wu, L.; Shan, L. Improved Rice Pest Recognition Based on SSD Network Model. *J. Zhengzhou Univ. (Nat. Sci. Ed.)* **2020**, *52*, 49–54. [CrossRef]
16. Liu, Q.; Yan, Z.; Wang, F.; Ding, C. Research on Object Detection Algorithm for Small Object of Pests Based on YOLOv3. In Proceedings of the 2021 International Conference on Computer Information Science and Artificial Intelligence (CISAI), Kunming, China, 17–19 September 2021; pp. 14–18.
17. Zhang, W.; Chen, S.; Wang, Y.; Tie, J.; Ding, J.; Li, M.; Li, C.; Su, X.; Gan, Y. Identification of Lasioderma serricorne in Tobacco Leaf Raw Materials Based on Improved YOLOv3 Algorithm. *J. Henan Agric. Sci.* **2023**, *52*, 157–166. [CrossRef]
18. Li, J.; Li, J.; Zhao, X.; Su, X.; Wu, W. Lightweight Detection Networks for Tea Bud on Complex Agricultural Environment via Improved YOLO V4. *Comput. Electron. Agric.* **2023**, *211*, 107955. [CrossRef]
19. Zhang, C.; Kang, F.; Wang, Y. An Improved Apple Object Detection Method Based on Lightweight YOLOv4 in Complex Backgrounds. *Remote Sens.* **2022**, *14*, 4150. [CrossRef]
20. Zhang, M.; Xu, S.; Song, W.; He, Q.; Wei, Q. Lightweight Underwater Object Detection Based on YOLO v4 and Multi-Scale Attentional Feature Fusion. *Remote Sens.* **2021**, *13*, 4706. [CrossRef]
21. Sun, Y.; Zhang, D.; Guo, X.; Yang, H. Lightweight Algorithm for Apple Detection Based on an Improved YOLOv5 Model. *Plants* **2023**, *12*, 3032. [CrossRef] [PubMed]
22. Kang, J.; Zhang, W.; Xia, Y.; Liu, W. A Study on Maize Leaf Pest and Disease Detection Model Based on Attention and Multi-Scale Features. *Appl. Sci.* **2023**, *13*, 10441. [CrossRef]
23. Wu, X.; Zhan, C.; Lai, Y.K.; Cheng, M.M.; Yang, J. IP102: A Large-scale Benchmark Dataset for Insect Pest Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8779–8788.
24. Russell, B.C.; Torralba, A.; Murphy, K.P.; Freeman, W.T. LabelMe: A Database and Web-Based Tool for Image Annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173. [CrossRef]
25. Shorten, C.; Khoshgoftaar, T.M. A Survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
26. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
27. Reis, D.; Kupec, J.; Hong, J.; Daoudi, A. Real-Time Flying Object Detection with YOLOv8. *arXiv* **2023**, arXiv:2305.09972v1.
28. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
29. Terven, J.; Cordova-Esparza, D. A Comprehensive Review of YOLO: From YOLOv1 and Beyond. *arXiv* **2023**, arXiv:2304.00501.
30. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
31. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
32. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
33. Yang, G.; Lei, J.; Zhu, Z.; Cheng, S.; Feng, Z.; Liang, R. AFPN: Asymptotic Feature Pyramid Network for Object Detection. In Proceedings of the 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Honolulu, HI, USA, 1–4 October 2023.
34. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
35. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-Neck by GSConv: A Better Design Paradigm of Detector Architectures for Autonomous Vehicles. *arXiv* **2022**, arXiv:2206.02424.
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385v1.
37. Hassanin, M.; Anwar, S.; Radwan, I.; Khan, F.S.; Mian, A. Visual Attention Methods in Deep Learning: An In-Depth Survey. *arXiv* **2022**, arXiv:2204.07756.
38. Yang, L.; Zhang, R.-Y.; Li, L.; Xie, X. SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021.
39. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; NanoCode012; Kwon, Y.; Michael, K.; Tao, X.; Fang, J.; Imyhxy; et al. Ultralytics. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 9 November 2022).
40. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
41. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.